REMARKS

Claims 1-21 are pending and stand rejected on various prior art grounds as either anticipated by Campbell or obvious in view of a suggested combination of Campbell and Chilimbi. Applicants traverse these rejections for the following reasons. Reconsideration is respectfully requested.

The present application is generally related to techniques that can automatically monitor performance of a memory heap, determine if the memory heap has delinquent regions, and instead of merely avoiding that delinquent memory region, automatically perform memory management routines to optimize that memory heap. Claim 1, for example, recites:

> 1. An article comprising a machine-accessible medium having stored thereon instructions that, when executed by a machine, cause the machine to:
>
> obtain, from a performance monitor, performance data for a memory heap having a plurality of memory regions;
>
> based on the performance data, determine if at least one of the plurality memory regions is a delinquent region; and
>
> in response to a determination that at least one of the plurality of memory regions is a delinquent region, execute a memory management routine to optimize that delinquent region of the memory heap.

In contrast to memory heap techniques, Campbell relates to a system for managing the cache. The Campbell system selectively assigns data elements to cache locations based on memory performance, whereas the present application identifies problem memory heap areas and optimizes them for continued storage of data. There is a difference between cache allocation and heap memory optimization. A cache merely holds (possible coherent) copies of data in main (or heap) memory, thus permitting faster access to the data. Where in the cache hierarchy these copies are held is the focus of Campbell's work; but Campbell does not seek to fix problems at the main memory level, where memory problems would continue to persist.

Campbell actually uses rule-based cache assignment in lieu of optimization, instead diverting data from a problem cache level to another cache. To the extent the Campbell system is managing cache allocation in response to monitored parameters, the system is merely determining where a data element should be cached based, not how that data is arranged in main (or heap) memory.

Campbell is concerned about which level of the cache hierarchy to place data (cache lines) from memory, while the present application describes, *inter alia*, techniques for how to arrange data (objects) in memory and how to focus the optimization techniques, such as garbage collection, to better arrange data. With techniques such as those described, for example, one may optimize the memory heap to better utilize the limited DTLB and cache capacity. Campbell in no way addresses DTLB issues or how to organize data to better utilize the cache. Instead, Campbell is focused on where in the cache hierarchy data should be placed.

There are two parts of Campbell's rule-based cache assignment. First, with respect to cache line allocation, Campbell does arguably perform cache line allocation in response to parameters that may indicate a delinquent memory, such as cache misses, but importantly instead of trying to correct that delinquent memory, Campbell merely avoids it by placing the cached data in another cache location. There is no teaching or suggestion of determining that a delinquent memory region exists and in response executing "a memory management routine <u>to optimize that region of the memory heap,</u>" as recited in claim 1.

Second, with respect to eviction (and applying the examiner's reading of Campbell), Campbell does describe removing data stored in a cache, but this is quite different than executing a memory management routine to optimize a delinquent region of the memory heap, as recited in claim 1.

As Campbell describes, Cache eviction occurs through couplings directly from the L1 cache to the eviction determining logic.

> Based on the cache misses between the first L1 cache 102 and the second l1 cache 103, counter 104 controls cache eviction location determining logic 105

> via coupling 115. when a cache line is evicted and first L1 cache 102 or second L1 cache 103 is needed to store additional data from processor 101, the existing data in first L1 cache 102 and second L1 cache 103 are evicted from their respective caches via couplings 113 and 114 through cache eviction location determining logic 105 and via couplings 116 and 117 to either of first L2 cache 106 or second L2 cache 107. Campbell 2:26-37.

Or cache eviction occurs through direct connection between the L1 and L2 caches.

> Alternatively, data may be evicted directly from first L1 cache 102 and second L1 cache 103 directly to the first L2 cache 106 and second L2 cache 107 via couplings 120 and 121 under the control of cache eviction location determining logic 105. Campbell 2:37-42.

Once data has been identified for cache eviction, instead of trying to fix the offending memory, Campbell assigns the data to a <u>new</u> cache line based on some thresholds. The system, for example, measures a threshold of cache misses by incrementing for misses in the first L1 cache and decrementing misses in the second L1 cache. Once either the positive threshold or the negative threshold is met,.eviction occurs.[1]

Campbell does not attempt any optimization of the actual memory region causing the cache miss. Indeed, Campbell does not even identify which portions of its caches have caused error. Campbell's decisions occur at the cache line or set level not the memory region level.

Unlike Campbell, the present application discloses various techniques for not only obtaining performance data for a memory heap and determining if one of a plurality of memory regions in the memory heap is a delinquent region, but also in response, executing a memory management routine to optimize that delinquent region of the memory heap. This optimization allows the memory heap to continue storing the stored data. The present application, for example, describes applying memory optimization techniques such as a garbage collection (e.g., a counting

[1] Allocation between the two L2 caches may occur based on the empty space or allocated space in each L2 cache. Campbell 3: 47-55. As shown in Figure 5, allocation can occur based on which cache as the fewest number of counted cache misses. Figure 6 shows that a counter is used to determine whether an allocation should go to the second cache, for example, when the counter is greater than a high threshold, to a first cache, for example, when the counter is less than a lower threshold, or to the either the first or second if no threshold is met.

collection, copy collection, slide compaction, generational collection, mark-sweep collection, beltway collection, oldest first collection, or a hybrid) in response to a determination that a memory region is delinquent.

In any event, it is clear from all of these discussions in Campbell, this reference merely describes a technique for rule-based assignment of data between caches. There is no teaching or suggestion of determining when a delinquent memory region exists and in response executing "a memory management routine to optimize that region of the memory heap," as recited in claim 1.

The anticipation rejections based on Campbell are traversed for at least the foregoing outlined reasons.

Applicants also respectfully submit that there is no teaching or suggestion rendering obvious the claimed subject matter. Although not raised with respect to claim 1, Chilimbi would not be combined with Campbell as the examiner has suggested with respect to other claims. As noted above Campbell applies a rule-based system for data assignment to address cache problems. Campbell would not be used with a system to optimize a memory region on the fly, as problems arise, because such optimization would, among other things, remove the need for the rule-based assignment and add unnecessary, time-consuming complexity.

The examiner points to no teaching, suggestion, or motivation in the prior art for combining Campbell with Chilimbi in the way suggested. The examiner merely says that at the time of the invention it would have been obvious to make the combination for the benefit of increasing overall cache response speed.

A closer look at Chilimbi further highlights however that there would be no teaching, suggestion, or motivation to combine the two references as suggested. Chilimbi is concerned with the internal structure of objects, and only indirectly with where in memory restructured objects are to placed. Campbell works at the cache line level without knowledge of objects. There is no suggestion of using Chilimbi's intra-object restructuring/splitting in Campbell's cache placement approach. Either way, there would be no teaching of how objects are arranged relative to each other, i.e., an inter-object approach.

For the foregoing reasons, applicants respectfully submit that claim 1 and claims 2 - 12 depending therefrom are in condition for immediate allowance.

With respect to the rejection of method claim 14, applicants reiterate the remarks above. Furthermore, applicants note that Campbell teaches a rule-based cache allocation system, which serves to evict data from a cache, but which does not optimize that cache in a way that it may still be used with the previously stored data. Campbell uses eviction to remove the data entirely, to be overwritten with new data. This overwriting without optimization could result in newly-stored data experiencing the same cache problems as the previously-stored data. Campbell does not teach a method that comprises optimizing a memory heap, for example, through a garbage collection routine like sliding compaction, that allows the stored memory to be retained in the memory region.

For the foregoing reasons, the rejections of claim independent 14 and the claims depending therefrom are traversed. Claims 14 – 18 are in condition for immediate allowance.

The applicants reiterate the remarks above in traversing the rejection of the remaining independent claim, claim 19.

For similar reasons to those outlined above, applicants respectfully submit that the subject matter of claim 19 is neither taught nor suggested by the art of record. Claim 19 and claims 20 and 21 depending therefrom are in condition for allowance.

In view of the above amendment, applicants believe the pending application is in condition for allowance.

Dated: July 17, 2006

Respectfully submitted,

By____________________
Paul B. Stephens
Registration No.: 47,970
MARSHALL, GERSTEIN & BORUN LLP
233 S. Wacker Drive, Suite 6300
Sears Tower
Chicago, Illinois 60606-6357
(312) 474-6300
Attorney for Applicant